



LIBRARY
OF THE
UNIVERSITY
OF ILLINOIS

510.84

Il6r

no. 228-236

cop. 2

The person charging this material is responsible for its return to the library from which it was withdrawn on or before the **Latest Date** stamped below.

Theft, mutilation, and underlining of books are reasons for disciplinary action and may result in dismissal from the University.

UNIVERSITY OF ILLINOIS LIBRARY AT URBANA-CHAMPAIGN

MAY 31 1979
MAY 3 REC'D
JUN 7 1980
MAY 13 REC'D

MAY 04 1998

264
234
.2
Report No. 234

MAJOR AND MINOR THRESHOLD FUNCTIONS OF SIX VARIABLES

by

John A. Rohr

June 17, 1967

THE LIBRARY OF THE
DEC 21 1967
UNIVERSITY OF ILLINOIS



DEPARTMENT OF COMPUTER SCIENCE · UNIVERSITY OF ILLINOIS · URBANA, ILLINOIS

Report No. 234

MAJOR AND MINOR THRESHOLD FUNCTIONS OF SIX VARIABLES*

by

John A. Rohr

June 17, 1967

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

* This work was supported in part by the University Research Board and was submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering, June, 1967.



Digitized by the Internet Archive
in 2013

<http://archive.org/details/majorminorthresh234rohr>

PREFACE

The investigation described herein was undertaken as a Master's thesis project under the direction of Dr. Saburo Muroga of the Department of Computer Science of the University of Illinois. Because of the extensive length of the results, they are not reproduced in detail, but are summarized in the text. The author has retained two copies of the complete results of the program. The author appreciates the guidance given by Dr. Muroga and also the help of Mr. Teiichi Tsuboi in preparing the generation program. Also, the support by the University Research Board for the computer time used by this investigation is greatly appreciated.

TABLE OF CONTENTS

	Page
I. INTRODUCTION.....	1
II. BASIC IDEAS OF THRESHOLD LOGIC.....	2
III. DEFINITION OF THE PROBLEM.....	4
IV. DESCRIPTION OF METHODS USED.....	6
V. DESCRIPTION OF THE PROGRAMS.....	16
VI. SOLUTION OF THE PROBLEM.....	20
VII. CONCLUSION.....	26
BIBLIOGRAPHY.....	27

I. INTRODUCTION

Switching circuits based on the threshold and majority decision principles have recently been used in certain computers. The behavior of these circuits can be represented by a Boolean function whose value is 1 or 0 (corresponding to "on" and "off") depending on whether the weighted sum of the variables is or is not greater than a specified value called the threshold.

This investigation has two main objectives:

1. The tabulation of all the representative major and minor threshold functions of six and fewer variables; and
2. The comparison of the corresponding optimum structures obtained by minimizing
 - a. the threshold, and
 - b. the total input weight.

II. BASIC IDEAS OF THRESHOLD LOGIC

The terms "threshold function" and "majority function" can be used interchangeably in most cases. There is a one-to-one correspondence between threshold functions and majority functions. The definitions of a threshold function and a majority function are given here.

A "threshold function" of n variables is a Boolean function, each of whose variables $x_1, x_2, \dots, x_n$ can have the value 1 or 0 and whose functional value is

$$1 \text{ for } \sum_{i=1}^n w_i x_i \geq T$$

$$0 \text{ for } \sum_{i=1}^n w_i x_i \leq T - 1$$

where each w_i is a prescribed constant real number called a "coupling weight" associated with the variable x_i and T is a prescribed constant real number called a "threshold".¹

A "majority function" of n variables is a Boolean function whose constant y_0 is identically +1 or -1 and each of whose variables $y_1, y_2, \dots, y_n$ can have the value +1 or -1 and whose functional value is

$$+1 \text{ for } \sum_{i=0}^n w_i y_i \geq +1$$

$$-1 \text{ for } \sum_{i=0}^n w_i y_i \leq -1$$

where each w_i is a prescribed constant real number called a "coupling weight" associated with the variable y_i (or constant y_0).

1. Saburo Muroga, Threshold Logic - Lecture Notes for EE 497 and EE 498, Department of Computer Science, University of Illinois, Urbana, Illinois, 1966, p. II-26.

The "total input weight" is $W = \sum_{i=0}^n |w_i|$.²

The set of $n + 1$ real numbers which specifies the behavior of a threshold function $(w_1, w_2, \dots, w_n; T)$ or a majority function $(w_1, w_2, \dots, w_n; w_0 y_0)$ is called the "structure" of the function. For any given function, the relationship between T and w_0 is expressed by

$$2T = \sum_{i=1}^n w_i - w_0 y_0 + 1.$$
³

2. Muroga, p. II-28.
3. Muroga, p. II-28.

III. DEFINITION OF THE PROBLEM

The central problem of this investigation is the determination of the optimum structures for the representative major and minor threshold functions of six and fewer variables. Because there are 14,487,040 threshold functions of six variables,⁴ only NPN-representative functions are considered. NPN-equivalent functions are those which can be obtained from the original function by any combination of 1. complementation of one or more of the variables, 2. permutation of the variables, and 3. complementation of the function. Thus the restrictions here on the representative functions are that all the variables must appear uncomplemented (i.e. the function is positive) and the weights must be in nonascending order (i.e. $w_1 \geq w_2 \geq \dots \geq w_n$).

The optimum structure for a threshold function can be determined by linear programming. Either the threshold or the total input weight may be minimized. For minor and self-dual threshold functions, these problems are equivalent. For major threshold functions however, these problems are not equivalent. This can be proved by considering the conversion formula

$$2T = W - w_0 y_0 + 1 \quad \text{where } W = \sum_{i=0}^n |w_i|.$$

For self-dual threshold functions $w_0 = 0$ and for minor threshold functions $y_0 = -1$. In either case $2T = W + 1$ so T differs from W only by multiplicative and additive constants. Thus minimization

4. Muroga, p. IX-31.

of T is equivalent to minimization of W for minor and self-dual threshold functions. For major threshold functions however, $y_0 = +1$ and $2T = W - 2w_0 + 1$ so minimization of T may yield different results than those obtained by minimization of W .

The formal statements of the two linear programming problems are given here.

For a positive threshold function $f(x_1, x_2, \dots, x_n)$, minimize T under the constraints

$$w_i \geq 0 \text{ for } i = 1, 2, \dots, n$$

$$T \geq 0$$

and
$$\sum_{i=1}^n w_i x_i^j \geq T \text{ for } f(x_1^j, x_2^j, \dots, x_n^j) = 1 \text{ (} j = 1, 2, \dots, B \text{)}$$

$$\sum_{i=1}^n w_i x_i^j \leq T-1 \text{ for } f(x_1^j, x_2^j, \dots, x_n^j) = 0 \text{ (} j = B+1, \dots, m \text{)}.^5$$

For a positive majority function $f(y_1, y_2, \dots, y_n)$,

minimize $W = \sum_{i=0}^n |w_i|$ under the constraints

$$w_i \geq 0 \text{ for } i = 0, 1, 2, \dots, n$$

and
$$\sum_{i=0}^n w_i y_i^j \geq +1 \text{ for } f(y_1^j, \dots, y_n^j) = 1 \text{ (} j = 1, 2, \dots, B \text{)}$$

$$\sum_{i=0}^n w_i y_i^j \leq -1 \text{ for } f(y_1^j, \dots, y_n^j) = 0 \text{ (} j = B+1, \dots, m \text{)}.^6$$

In this investigation both problems are solved and corresponding optimum structures are then compared.

5. Muroga, p. III-21.

6. Muroga, p. III-20.

IV. DESCRIPTION OF METHODS USED

The procedure used to accomplish the tabulation of the major and minor threshold function representatives is actually a combination of three separate algorithms. The first algorithm generates the truth table for each of the seven-variable, self-dual threshold function representatives. The second algorithm transforms each of the above functions into a set of six-variable major and minor functions. The third algorithm tests each of the six-variable functions to check that they truly are threshold functions and gives the optimum structure for each.

The method used to generate the seven-variable, self-dual threshold function representatives is essentially that used and described by Winder.⁷ Using his method, the truth tables for these functions are generated sequentially. Because these functions are self-dual, only half of the table for each function actually needs to be stored in the computer memory. Specifically, only the lower half, the portion having minterms with $x_1 = 0$, is explicitly stored. These half-tables are used as input to the transition steps.

For the following discussion, let each function be specified by a truth table of minterms having these minterms listed by descending binary value. Thus a "higher" minterm will be considered to be one having greater binary value.

7. R. O. Winder, "Enumeration of Seven-Argument Threshold Functions", IEEE Transactions on Electronic Computers, Vol. EC-14, No. 3, June, 1965, pp. 315 - 325.

For example, the minterm 1011 is higher than the minterm 1001. A minterm having the value 0 will be called a 0-minterm; if it is also an extremal it will be called a 0-extremal. A minterm having the value 1 will be called a 1-minterm; if it is also an extremal it will be called a 1-extremal. Prior to the main computation, a "choosing extremals table" was prepared. This table has an entry for each possible ordered pair of minterms. If both minterms are assumed to be 0-minterms, and if the second minterm is then "covered" by the first, then the entry in the "choosing extremals table" corresponding to this pair is a zero. Otherwise the table entry is a blank. The second minterm is "covered" by the first if no variable of the first is zero when the corresponding variable of the second is one (e.g. 1100 covers 1000 but does not cover 1010). This "choosing extremals table" is similar to the "choosing zero table" used in Winder's algorithm.⁸

The first step of the transition process is essentially the same as Processes A and B described by Muroga, Toda, and Takasu.⁹ This process designates extremals of the functions generated by the first algorithm. First all the 0-minterms for a function are considered. Beginning with the highest such 0-minterm, it is marked as a 0-extremal. All 0-minterms which are covered by this 0-extremal are then eliminated from

8. Winder, p. 319.

9. Saburo Muroga, Iwao Toda, and Satoru Takasu, "Theory of Majority Decision Elements", Journal of the Franklin Institute, Vol. 271, No. 5, May, 1961, p. 405.

further consideration. The process is then repeated using the highest remaining 0-minterm. When all 0-minterms have been marked as extremals or eliminated, the process is then terminated. Then in an analogous manner, but proceeding from the lowest 1-minterm, the process is repeated for 1-minterms. Thus the half-truth-table generated by the first algorithm has the extremals explicitly designated.

The next step of the transition process is to create a mirror image of the existing half-truth-table and place this above the original half-table. Thus the entire truth table for the seven-variable, self-dual threshold function is explicitly stored in the computer memory, complete with the extremal designations previously generated.

Finally, from the complete truth table, the extremals for the six-variable major and minor functions are selected. Beginning with the first variable, the extremals for the major function are selected first. All extremals having $x_1 = 1$ are selected. These extremals are then used as input to the third algorithm, the linear program test-synthesis. The results from this linear program are stored in the computer memory for later sorting and tabulation. Then all the extremals in the truth table which have $x_1 = 0$ are selected for the minor function. Again a linear program is run and the results are stored in the computer memory. Then the process of selecting extremals for major and minor functions is repeated for all the remaining variables of the original seven-variable function.

When all possible major and minor function extremals have been selected from a seven-variable, self-dual function, the generation algorithm is called to produce another such function. When the generation algorithm terminates, the previously stored linear program results are evaluated.

The generation program produces all major and minor threshold functions of six and fewer variables. The are intermixed as they are generated. For the tabulation, first the two-variable functions are selected from the stored list. These are ordered according to structure and then listed. Then the procedure is repeated for the three-, four-, five-, and six-variable functions.

The third algorithm of the investigation, the linear program test-synthesis previously mentioned, is essentially the same as the one described by Muroga, Toda, and Takasu.¹⁰ Using extremals generated by the transition steps, inequalities are generated and a linear program is run to determine if the function is truly a threshold function, and if it is, what the optimum structure is.

Thus the first algorithm generates the seven-variable, self-dual threshold function representatives, the transition algorithm transforms these functions to extremals for six-variable major and minor functions, and the third algorithm checks to be sure that the functions generated are indeed threshold functions, and determines the optimum structures.

10. Muroga, Toda, and Takasu, pp. 408 - 411.

By minor modifications of the section of the generation program which generates inequalities and the objective function for the linear program, the optimum structures corresponding to minimized total input weight for all major and minor threshold functions of six and fewer variables were obtained in a manner exactly parallel to that described above for optimum structures corresponding to minimized threshold.

Both programs were then modified to write their results onto magnetic tape. A program was then written and used which automatically compared these tapes to determine threshold functions having different optimum structures corresponding to minimized threshold and minimized total input weight.

As an example of the transition algorithm, consider the self-dual function of five variables shown in Table 1. Winder uses this function to illustrate his algorithm.¹¹ Table 2 shows the "choosing extremals table" used for this function. Table 3 shows the designation of extremals and which extremals cover which minterms. Table 4 shows the complete truth table for the self-dual function, complete with the extremal designation. Finally, Table 5 lists the extremals for the five major functions and five minor functions which result from the self-dual function.

11. Winder, p. 320.

Minterm	Value
01111	1
01110	1
01101	1
01100	0
01011	0
01010	0
01001	0
01000	0
00111	0
00110	0
00101	0
00100	0
00011	0
00010	0
00001	0
00000	0

Table 1. Example Function As Generated.

Minterm	Index																
		1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
01111	1	0															
01110	2	0	0														
01101	3	0		0													
01100	4	0	0	0	0												
01011	5	0				0											
01010	6	0	0			0	0										
01001	7	0		0		0		0									
01000	8	0	0	0	0	0	0	0	0								
00111	9	0								0							
00110	10	0	0							0	0						
00101	11	0		0						0		0					
00100	12	0	0	0	0					0	0	0	0				
00011	13	0				0				0				0			
00010	14	0	0			0	0			0	0			0	0		
00001	15	0		0		0		0		0		0		0		0	
00000	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 2. Choosing Extremals Table For Example Function.

Minterm	Value	Extremals	Extremals Which Cover Minterms				
			01110	01101	01100	01011	00111
01111	1		x	x			
01110	1	*	x				
01101	1	*		x			
01100	0	*			x		
01011	0	*				x	
01010	0					x	
01001	0					x	
01000	0				x	x	
00111	0	*					x
00110	0						x
00101	0						x
00100	0				x		x
00011	0					x	x
00010	0					x	x
00001	0					x	x
00000	0				x	x	x

Note: x signifies which minterms are covered in each column.

Table 3. Example Function With Extremals Designated.

Minterm	Value	Extremals
11111	1	
11110	0	
11101	1	
11100	1	
11011	1	
11010	1	
11001	1	
11000	1	*
10111	1	
10110	1	
10101	1	
10100	1	*
10011	1	*
10010	0	*
10001	0	*
10000	0	
01111	1	
01110	1	*
01101	1	*
01100	0	*
01011	0	*
01010	0	
01001	0	
01000	0	
00111	0	*
00110	0	
00101	0	
00100	0	
00011	0	
00010	0	
00001	0	
00000	0	

Table 4. Complete Truth Table For Example Function.

Variable	Major Functions		Minor Functions	
	Extremal	Value	Extremal	Value
x_1	1000	1	1110	1
	0100	1	1101	1
	0011	1	1100	0
	0010	0	1011	0
	0001	0	0111	0
x_2	1000	1	1100	1
	0110	1	1011	1
	0101	1	1010	0
	0100	0	1001	0
	0011	0	0111	0
x_3	1000	1	1100	1
	0110	1	1011	1
	0101	1	1010	0
	0100	0	1001	0
	0011	0	0111	0
x_4	1001	1	1100	1
	1000	0	1010	1
	0110	1	1001	0
	0101	0	0111	1
	0011	0	0110	0
x_5	1001	1	1100	1
	1000	0	1010	1
	0110	1	1001	0
	0101	0	0111	1
	0011	0	0110	0

Table 5. Extremals For Major And Minor Functions
Derived From Example Self-Dual Function.

V. DESCRIPTION OF THE PROGRAMS

The main program used for this investigation consists of two parts: a generation program which generates all threshold function candidates and a linear program subroutine which determines whether or not a candidate is actually a threshold function, and if so, what its optimum structure is. As originally written, the program minimizes the threshold. With minor modifications it can be used to minimize the total input weight. Included in the generation program are auxiliary routines which print the extremals and the "choosing zero table" and the "choosing extremals table". These routines are called by dummy subroutines. Throughout the entire program, as many tricks as possible are used to save time. Even a saving of a few microseconds can be significant because most commands are executed many times: some more than a million times during a complete run.

As originally written, the program specified which structures were definitely unique, but it did nothing with those which were possibly nonunique. By modifying the program to check simple cases and print pertinent data from the linear program for all remaining cases, uniqueness was checked for all structures.

A separate program was written to compare optimum structures obtained for the same function by first minimizing the threshold and then minimizing the total input weight. In order to accomplish

this the main programs were modified to write their results onto magnetic tape. The comparison program then read these tapes simultaneously and compared the optimum structures obtained for the two problems. Any functions whose optimum structures for minimized threshold differed from that for minimized total input weight were printed for further consideration. Actually, the results shown in Table 6 were not obtained until uniqueness of optimum structures was checked.

For testing purposes all programs were written so that they would work the two-variable case first, then do the three-variable case and continue on through the six-variable case. Provisions were included so that the program could be terminated after any particular case by including one data card which specified the number of variables for the final case.

The generation and transition processes are relatively straightforward. It was originally intended to use Processes C and D of Muroga, Toda, and Takasu¹² to eliminate all redundant extremals, but because of the speed of the program this was unnecessary and was never included in the program.

The algorithm used in the linear program is the simplex method. The linear program subroutine treats all numbers as rational numbers, keeping the numerator and denominator as separate 35-bit integers. This is necessary to eliminate rounding of numerical results so that the weights which result from the linear program can be checked

12. Muroga, Toda, and Takasu, p. 407.

to see that they are integral. To keep the numbers within range, two subprograms were included in the linear program subroutine. The first subprogram attempts all indicated divisions as they occur. If the result is integral, the numerator of the number is set equal to the quotient and the denominator is set to one. Otherwise the original rational number is left as the result. The second subprogram tries, after each iteration after the third for each problem, to reduce each numerator and its corresponding denominator by dividing each by the greatest common divisor of the two, if it is greater than one. These subprograms succeeded in keeping the numbers within range, but it appears that the second one was really unnecessary: the largest numerator that appeared was 512 and the largest denominator was 1024.

All programs were written for an IBM 7094 computer using SCATRE, an assembly language for the machine. The basic main program was written to solve the threshold minimization problem. The generation program consists of 1661 cards and the linear program subroutine has 603 cards. Modifications to the program for the total input weight minimization problem bring the totals to 1691 and 604 respectively. Cards inserted to solve the uniqueness problem number 83 for threshold minimization and 66 for total input weight minimization. (The reason that more are required for the former is that threshold functions which have a single literal prime implicant, e.g. $x_1 \vee x_2 x_3$ are known to have multiple optimum structures and are to be excluded from consideration to save computation time.) The modification to the program which

causes the results to be written onto magnetic tape requires 45 cards for the threshold minimization problem and 42 for the total input weight minimization problem.

The complete tabulation of the optimum structures of the major and minor threshold functions of two to six variables requires 2366 linear programming problems, a large sort, the listing, and collection of pertinent statistics. The time for a complete run is approximately 10.5 minutes.

VI. SOLUTION OF THE PROBLEM

This investigation has produced a list of the NPN-representative major and minor threshold functions of two to six variables. The trivial case of one variable was not included. Muroga, Toda, and Kondo¹³ have listed the minor and self-dual majority functions for the same cases. Winder¹⁴ has listed the seven-variable, self-dual majority functions which are the same as those used here to generate the six-variable major and minor functions. The results of this investigation agree with both tabulations.

The major new result of this investigation is the discovery of six major, six-variable functions, listed in Table 6, which have different optimum structures corresponding to minimized threshold and minimized total input weight. Threshold functions which include single literal prime implicants are known to have different structures corresponding to minimized threshold and minimized total input weight, and thus were eliminated from consideration. Self-dual and minor threshold functions are known to have the same structure whether the threshold or the total input weight is minimized because the linear programming problems are equivalent. For major threshold functions however, it was not known if the resulting optimum structures were identical for functions which do not have any single literal prime implicants.

-
13. Saburo Muroga, Iwao Toda, and M. Kondo, "Majority Decision Functions of up to Six Variables", Mathematics of Computation, Vol. XVI, No. 80, October, 1962, pp. 459 - 472.
14. R. O. Winder, "Threshold Functions Through $n = 7$ ", AFCRL Report 64-925 under contract AF 19(604) with RCA Laboratories, October, 1964.

The discovery of six special cases where the optimum structures are different shows that the optimum structures for major threshold functions corresponding to minimized threshold and minimized total input weight are not always identical.

Some statistics collected by the programs are presented in Tables 7 and 8. Table 7 lists statistics pertaining to the tabulation. The threshold minimization statistics are for all major and minor threshold function representatives of the specified number of variables. The total input weight minimization statistics pertain only to major and self-dual functions. Minor functions were not included because the structure for each is identical with that of the corresponding major function except that the sign of the constant input is negative. Because of the way the program was written, self-dual functions are included in the total input weight minimization tabulation.

It can be seen from Table 7 that for the two- and three-variable cases the number of iterations is about one more than the number of extremals. As the number of extremals begins to grow however, the number of iterations grows also, but not nearly so fast as the number of extremals. Statistics concerning the maximum size of the largest and smallest weight and the threshold and constant weight are listed for general interest and for checking purposes. Table 8 gives statistics which show the results of the subprograms of the linear program subroutine. Averages are over the nonzero cases, i.e. the cases where at least one operation was performed. The division program was more successful in the threshold minimization case, but in both cases usually was able

to perform the division. The reduction program was more successful in the total input weight minimization case, but in neither case was the success outstanding. It appears that the program to divide out common factors in the numerator and denominator may actually have been unnecessary.

The complete list of results consists of about 200 pages of computer output for each case. These results are too extensive to reproduce in detail and hence have been summarized in this section.

No. Function

1. $x_1x_2 \vee x_1x_3 \vee x_1x_4 \vee x_1x_5 \vee x_1x_6 \vee x_2x_3 \vee x_2x_4x_5 \vee$
 $x_2x_4x_6 \vee x_2x_5x_6 \vee x_3x_4x_5 \vee x_3x_4x_6 \vee x_3x_5x_6 \vee x_4x_5x_6$
2. $x_1x_2 \vee x_1x_3 \vee x_1x_4 \vee x_1x_5 \vee x_1x_6 \vee x_2x_3x_4 \vee x_2x_3x_5 \vee$
 $x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6 \vee x_2x_5x_6 \vee x_3x_4x_5 \vee x_3x_4x_6 \vee x_3x_5x_6$
3. $x_1x_2 \vee x_1x_3 \vee x_1x_4 \vee x_1x_5 \vee x_1x_6 \vee x_2x_3 \vee x_2x_4 \vee$
 $x_2x_5x_6 \vee x_3x_4x_5 \vee x_3x_4x_6 \vee x_3x_5x_6 \vee x_4x_5x_6$
4. $x_1x_2 \vee x_1x_3 \vee x_1x_4 \vee x_1x_5 \vee x_1x_6 \vee x_2x_3x_4 \vee x_2x_3x_5 \vee$
 $x_2x_3x_6 \vee x_2x_4x_5 \vee x_2x_4x_6 \vee x_2x_5x_6 \vee x_3x_4x_5 \vee x_3x_4x_6$
5. $x_1x_2 \vee x_1x_3 \vee x_1x_4 \vee x_1x_5 \vee x_1x_6 \vee x_2x_3 \vee x_2x_4 \vee$
 $x_2x_5x_6 \vee x_3x_4x_5 \vee x_3x_4x_6 \vee x_3x_5x_6$
6. $x_1x_2 \vee x_1x_3 \vee x_1x_4 \vee x_1x_5 \vee x_1x_6 \vee x_2x_3 \vee x_2x_4x_5 \vee$
 $x_2x_4x_6 \vee x_2x_5x_6 \vee x_3x_4x_5 \vee x_3x_4x_6$

Optimum Structures

No.	Minimized Threshold								Minimized Total Input Weight							
	W	w ₁	w ₂	w ₃	w ₄	w ₅	w ₆	T	W	w ₁	w ₂	w ₃	w ₄	w ₅	w ₆	w ₀ y ₀
1.	16-17	4-5	3	3	2	2	2	6	21	4	3	3	2	2	2	+5
2.	17-18	5-6	3	3	2	2	2	7	21	5	3	3	2	2	2	+4
3.	19-20	5-6	4	3	3	2	2	7	25	5	4	3	3	2	2	+6
4.	20-21	6-7	4	3	3	2	2	8	25	6	4	3	3	2	2	+5
5.	22-23	6-7	5	4	3	2	2	8	29	6	5	4	3	2	2	+7
6.	23-24	7-8	5	4	3	2	2	9	29	7	5	4	3	2	2	+6

Table 6. Functions Having Different Structures Corresponding To Minimized Threshold And Minimized Total Input Weight.

		True Number of Variables				
	Totals	2	3	4	5	6
Threshold Minimization						
Number of Functions	1090	2	4	16	88	980
Average Number of Extremals	14.3	3.0	4.0	6.1	9.3	14.9
Maximum Number of Extremals		3.0	4.0	10.0	15.0	35.0
Average Number of Iterations	9.6	4.0	5.0	6.0	7.6	9.8
Maximum Number of Iterations		4.0	5.0	6.0	11.0	16.0
Maximum Largest Weight (w_1)		1	2	3	5	9
Maximum Smallest Weight (w_n)		1	1	1	2	3
Maximum Threshold (T)		2	3	5	9	18
Total Input Weight Minimization*						
Number of Functions	565	1	3	9	48	504
Average Number of Extremals	14.0	3.0	3.7	5.9	9.1	14.7
Maximum Number of Extremals		3.0	4.0	10.0	15.0	35.0
Average Number of Iterations	9.5	4.0	4.7	5.9	7.5	9.8
Maximum Number of Iterations		4.0	5.0	6.0	10.0	15.0
;						
Maximum Largest Weight (w_1)		1	2	3	5	9
Maximum Smallest Weight (w_n)		1	1	1	2	3
Maximum Constant Weight (w_0)		1	2	3	5	9

*Self-dual and major functions are actually tabulated.

Table 7. Tabulation Statistics.

		True Number of Variables				
	Totals	2	3	4	5	6
Threshold Minimization						
Average Division Attempts	739.1	84.0	130.0	224.5	387.5	794.6
Maximum Division Attempts		87.0	135.0	312.0	762.0	2832.0
Average Division Success (%)	97.0	100.0	100.0	100.0	99.8	96.6
Maximum Division Success (%)		100.0	100.0	100.0	100.0	100.0
Average Reduction Attempts	18.8	0.0	0.0	0.0	6.0	18.9
Maximum Reduction Attempts		0.0	0.0	0.0	6.0	223.0
Average Reduction Success (%)	55.3	0.0	0.0	0.0	50.0	55.3
Maximum Reduction Success (%)		0.0	0.0	0.0	50.0	100.0
Total Input Weight Minimization*						
Average Division Attempts	1018.0	100.0	133.3	275.2	509.5	1108.8
Maximum Division Attempts		100.0	169.0	378.0	1029.0	4598.0
Average Division Success (%)	68.7	75.0	72.5	70.7	68.4	68.6
Maximum Division Success (%)		75.0	75.0	79.6	81.0	84.4
Average Reduction Attempts	13.4	0.0	1.0	1.5	3.7	14.7
Maximum Reduction Attempts		0.0	1.0	3.0	20.0	357.0
Average Reduction Success (%)	99.7	0.0	100.0	100.0	100.0	99.7
Maximum Reduction Success (%)		0.0	100.0	100.0	100.0	100.0

*Self-dual and major functions are actually tabulated.

Table 8. Success Of Linear Program Subprograms.

VII. CONCLUSION

The tabulation of the optimum structures for all representative major and minor threshold functions of six and fewer variables has been achieved by writing a program which consists of three basic sections: the first section generates the seven-variable, self-dual threshold function representatives; the second section transforms these functions into extremals and then inequalities for six-variable major and minor threshold functions; and the third section determines the optimum structures for the functions. Further, six major, six-variable functions which do not have single literal prime implicants have been found which have different optimum structures corresponding to minimized threshold and minimized total input weight.

Thus the two main objectives of the investigation have been accomplished:

1. The tabulation has been completed; and
2. It has been shown that in most, but not all, cases
the optimum structures for major threshold
functions obtained by minimizing the threshold
and the total input weight are identical.

BIBLIOGRAPHY

1. Gass, Saul I., Linear Programming: Methods and Applications, McGraw-Hill Book Co., New York, 1958.
2. Hadley, G., Linear Programming, Addison-Wesley Publishing Co., Inc., Reading, Massachusetts, 1962.
3. Muroga, Saburo, Threshold Logic - Lecture Notes for EE 497 and EE 498, Department of Computer Science, University of Illinois, Urbana, Illinois 1966.
4. Muroga, Saburo, Iwao Toda, and M. Kondo, "Majority Decision Functions of up to Six Variables", Mathematics of Computation, Vol. XVI, No. 80, Oct., 1962, pp. 459 - 472.
5. Muroga, Saburo, Iwao Toda, and Satoru Takasu, "Theory of Majority Decision Elements", Journal of the Franklin Institute, Vol. 271, No. 5, May, 1961, pp. 376 - 418.
6. Winder, R. O., "Enumeration of Seven-Argument Threshold Functions", IEEE Transactions on Electronic Computers, Vol. EC-14, No. 3, June, 1965, pp. 315 - 325.
7. Winder, R. O., "Threshold Functions Through $n = 7$ ", AFCRL Report 64-925 under contract AF 19(604) with RCA Laboratories, October, 1964.

JAN 8 1968

2
1938

UNIVERSITY OF ILLINOIS-URBANA



3 0112 039801896